**НАВИГАЦИЯ**

|  |  |
| --- | --- |
| pwd | print working directory  покажи, в какой я папке |
| ls | list directory contents  покажи файлы и папки в текущей папке; |
| ls -a | вывести расширенный список. В нём отобразятся все скрытые файлы, которые начинаются с символа . (например, файлы конфигурации). В том числе два особых файла . и .., которые обозначают текущую и родительскую директории. |
| ls ~ | выведет содержимое домашней директории вне зависимости от того, что показывает pwd |
| ls .. | покажет содержимое родительской директории |
| cd first-project | change directory  перейди в папку first-project; |
| cd first-project/html | перейди в папку html, которая находится в папке first-project; |
| $ cd "Фотографии с дня рождения" | если в названии папки есть пробелы, при вводе нужно использовать кавычки |
| cd .. | перейди на уровень выше, в родительскую папку; |
| cd ~ | перейди в домашнюю директорию (/Users/Username); |
| cd / | перейди в корневую директорию. |

**РАБОТА С ФАЙЛАМИ И ПАПКАМИ**

СОЗДАНИЕ

|  |  |
| --- | --- |
| touch index.html | создай файл index.html в текущей папке |
| touch index.html style.css script.js | если нужно создать сразу несколько файлов, можно напечатать их имена в одну строку через пробел |
| mkdir second-project | make directory  создай папку с именем second-project в текущей папке. |
| $ mkdir -p dir1/dir-inside/dir-deeper-inside | Можно создать целую структуру директорий одной командой с помощью флага -p.  # создали папку dir-deeper-inside в папке dir-inside, которая находится в папке dir1 |
| mkdir ~/my-git-projects | создаст папку my-git-projects внутри домашней директории |
| touch ../../file.txt | создаст файл file.txt на две папки выше по иерархии. Допустим, если вы находитесь в директории projects/git/hello, команда touch ../../file.txt создаст файл по такому пути: projects/file.txt. |

КОПИРОВАНИЕ И ПЕРЕМЕЩЕНИЕ

|  |  |
| --- | --- |
| cp что\_копируем куда\_копируем |  |
| cp index.html src/ | copy скопируй файл в другое место  # скопировали index.html в папку src |
| cp что\_копируем что\_копируем что\_копируем куда\_копируем | можно указать сразу несколько файлов |
| cp index.html style.css script.js src/ | # скопировали три файла (index.html, style.css и script.js) в папку src |
| mv file.txt ~/my-dir | move  перемести файл или папку в другое место. |

ЧТЕНИЕ

|  |  |
| --- | --- |
| cat file.txt | concatenate and print  распечатай содержимое текстового файла file.txt. |

УДАЛЕНИЕ

|  |  |
| --- | --- |
| rm about.html | remove  удали файл about.html; |
| rmdir images | remove directory  удали папку images; |
| rm -r second-project | remove + recursive  удали папку second-project и всё, что она содержит. |

**Полезные возможности**

Команды необязательно печатать и выполнять по очереди. Можно указать их списком — разделить двумя амперсандами (&&).

У консоли есть собственная память — буфер с несколькими последними командами. По ним можно перемещаться с помощью клавиш со стрелками вверх (↑) и вниз (↓).

Чтобы не вводить название файла или папки полностью, можно набрать первые символы имени и дважды нажать Tab. Если файл или папка есть в текущей директории, командная строка допишет путь сама.

Например, вы находитесь в папке dev. Начните вводить cd first и дважды нажмите Tab. Если папка first-project есть внутри dev, командная строка автоматически подставит её имя. Останется только нажать Enter.

**Настройка Git**

Сейчас вы работаете в одиночку, но в дальнейшем вам может понадобиться использовать Git в команде. Чтобы участникам проекта было понятно, кто и какие изменения вносил, нужно представиться и указать имя пользователя и адрес электронной почты.

**1. Настройте Git командами, которые мы показали в этом уроке.**

$ git config --global user.name "ваше имя или ник латиницей"

$ git config --global user.email ваша электронная почта

**2. Убедитесь, что данные сохранились, с помощью одной из двух команд.**

$ cat ~/.gitconfig

$ git config --list